# Multi-Hierarchy Interaction Control of a Redundant Robot Using Impedance Learning

Yiming Jiang[a,c], Chenguang Yang[b,*], Yaonan Wang[a], Zhaojie Ju[c], Yanan Li[d], Chun-Yi Su[e]

[a]*College of Electrical and Information Engineering, Hunan University, Changsha 410082, China*
[b]*Bristol Robotics Laboratory, University of the West of England, Bristol, UK*
[c]*School of Computing, University of Portsmouth, Portsmouth PO1 3HE, UK*
[d]*School of Engineering and Informatics, University of Sussex, Brighton, BN1 9RH, UK*
[e]*Gina Cody School of Engineering and Computer Science, Concordia University, Montreal, Quebec H3G 1M8, Canada*

**Abstract**

The control of robots with a compliant joint motion is important for reducing collision forces and improving safety during human robot interactions. In this paper, a multi-hierarchy control framework is proposed for the redundant robot to enable the robot end-effector to physically interact with the unknown environment, while providing compliance to the joint space motion. To this end, an impedance learning method is designed to iteratively update the stiffness and damping parameters of the end-effector with desired performance. In addition, based on a null space projection technique, an extra low stiffness impedance controller is included to improve compliant joint motion behaviour when interaction forces are acted on the robot body. With an adaptive disturbance observer, the proposed controller can achieve satisfactory performance of the end-effector control even with the external disturbances in the joint space. Experimental studies on a 7 DOF Sawyer robot show that the learning framework can not only update the target impedance model according to a given cost function, but also enhance the task performance when interaction forces are applied on the robot body.

*Keywords:* robot interaction control, impedance control, impedance parameters learning, null space, disturbance observer

## 1. Introduction

In recent years, the potential of physical interactions in human robot co-existent scenarios has elicited large interests in industry and academia, leading to the recent growth of a variety of research topics such as physical human-robot interactions (pHRI) [1, 2, 3, 4], human intention estimation [5, 6, 7], haptic identification [8, 9] and so forth. Traditionally, industrial robot are controlled based on a high stiffness to guarantee the

*Corresponding author
*Email address:* cyang@ieee.org (Chenguang Yang)

motion precision. Robots are fixed in cages to avoid any contacts with the surrounded people, because the rigid collision may lead to large forces and even cause damage to human. Hence, a critical issue of the controller design for human robot interaction is to provide compliance to guarantee the safety and the flexibility of the robot [10, 11, 12].

In general, the robot flexibility can be achieved by redundant degrees of freedom (DOF) robots [13, 14, 15, 16]. For example, in a human pick-and-place task, additional control objectives such as collision avoidance can be accomplished besides the main tasks. This is achieved by using the null space projection of the kinematically redundant manipulator, or so-called null motion [17]. In a pHRI scenario, however, it is likely that an interactive/external force is applied on the robot body unpredictably. In this case, the kinematic level control commands may fail to drive the robot immediately to move away from its current position. Therefore, it is desirable to have a multi-priority framework at the dynamic torque control level.

In [18], a null space based force control algorithm was designed according to the system dynamic property. In [19], by employing a minimal null-space projection, a velocity controller was developed to ensure the stability of the robot in an extended operational space. In [20], a compliant dynamic control approach was presented by introducing specific coordinates of the null motion. In the studies mentioned above, the dynamics of the whole robot body can be represented by two parts, the end-effector dynamics and the null motion dynamics. However, to design a null space velocity controller is challenging since it needs to build a map between the joint velocities and the null space velocities, whose inverse solution may not be unique.

On the other hand, impedance control has been widely used by actively regulating the dynamic behaviour between force and position at the interaction points [21]. A Cartesian impedance based torque control with a prescribed impedance model was proposed in [22]. A predefined passivity-based impedance model was developed for a flexible joint robot using the singular perturbation method [23]. Note that in the impedance controllers mentioned above, the impedance parameters were fixed with a specified impedance model. However, a predefined impedance model might be conservative in many scenarios, especially when interacting with an unknown environment. Without the knowledge of environment model, the design of impedance controller with appropriate control performance is challenging. Hence, it is necessary to learn proper impedance parameters as well as the reference trajectory to achieve desired interaction performance. A natural actor-critic learning method was employed to find the optimal impedance parameters in a contact task [24]. In [25], a policy improvement reinforcement learning based variable impedance control was developed to optimize a selected cost function with specified task performance. In [26], an admittance based control algorithm was proposed to enable the robot to follow the human movement by estimating the human motion intention.

Motivated by the above discussion, in this paper, we aim to develop a multi-hierarchy impedance learning control framework, which enables the robot end-effector to interact with unknown environments without using priori environmental information. The proposed control scheme is mainly developed based on a technique of null space projection, which divides the robot dynamics into two subsystems. Meanwhile, a minimal parametric set is employed to represent the null space motion by using a joint space decomposition approach. In comparison to our previous study in [27] where a PID-

like controller was used, the proposed scheme enables the robot to perform compliant joint movement when external forces are applied. Additionally, a learning framework is further developed to iteratively update the impedance parameters for the robot to interact with the unknown environment. The developed impedance learning framework is much simpler than the actor-critic based reinforcement learning method in [24, 25], and updates the stiffness and damping parameters simultaneously.

It should also be noted that uncertainties may inevitably exist in a robotic system due to the factors like external disturbances, measurement noises and frictions, which have a major influence on the control accuracy. Approximation-based techniques like neural network or fuzzy logic systems can be used to solve the problem [28, 29, 30, 31]. However, the increased computational cost may prevent the applications for real-time systems. Another widely used approach to address the external disturbances is adaptive disturbance observer (DOB) [32, 33, 34]. Comparing with neural network, the DOB is a more effective method for on-line applications. A DOB was designed to deal with the negative effects of the disturbances of a flapping wing micro aerial vehicle [35]. In [36], a nonlinear disturbance observer was designed for the estimation of unknown disturbances as well as unmeasured states. In [37], a high-order disturbance observer was used for a mobile wheeled inverted pendulum by using optimal gain matrices. The disturbance observers combining with sliding mode control were also designed for discrete-time systems [38, 39]. Inspired by the idea of DOB, in this paper, an adaptive DOB is further developed to compensate for the disturbances caused by system uncertainties and external forces during the motion of the robot. The proposed approach is able to enhance the tracking accuracy of the multi-hierarchal control scheme. The main contributions of the proposed control scheme are listed as below.

- Develop a multi-priority algorithm to enable the redundant robot to perform compliant joint movements without affecting the tracking control of the end-effector by using a null space projection technique.
- Enable the robot to interact with an unknown environment without using priori environmental information by employing an iterative impedance learning algorithm.
- Integrate an adaptive disturbance observer to compensate for the external disturbances caused by measurement noises, external forces and frictions. Therefore, the tracking accuracy of the proposed control scheme is guaranteed.

The rest of this paper is organized as follows. Preliminaries of the redundant robots and learning framework are given in Section 2. The control design procedure is presented in Section 3, where the control objectives and stability analysis are included. Experimental results are presented in Section 4 to testify the proposed impedance learning algorithm and the null-space motion controller. Conclusions are given in Section 5.

## 2. Preliminaries

### *2.1. System Description of Redundant Robots*

The robot system studied in this paper is a $n$ DOF redundant robot manipulator while the workspace of the manipulator is $m$ dimensional with $m < n$. Through the

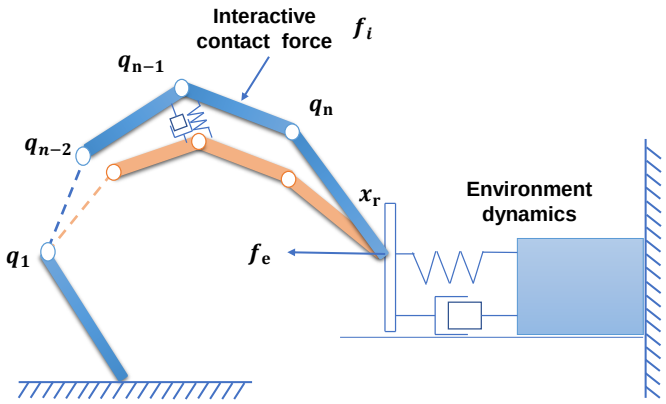


Figure 1: A robot interacts with an environment

well known forward kinematic equation, the relationship between end-effector velocity and joints velocity can be derived as follows

$$\dot{x} = J(q)\dot{q} \tag{1}$$

where $q \in \mathbb{R}^n$ and $\dot{q} \in \mathbb{R}^n$ denote the joint position and its first order derivative, respectively, $\dot{x}$ denotes the velocity of the end-effector, while $J(q) \in \mathbb{R}^{m\times n}$ denotes the Jacobian matrix of the robot.

For a redundant manipulator, the inverse solution of joint velocity is not unique due to the joint redundancy.

A general inverse solution of (1) can be expressed as follows,

$$\dot{q} = J^{\#}(q)\dot{x} + N\xi \tag{2}$$

where $J^{\#}(q) \in \mathbb{R}^{n\times m}$ is a generalized inverse satisfying $J = JJ^{\#}J$, $N = I - J^{\#}J$, and $\xi \in \mathbb{R}^n$ is an arbitrarily selected vector. Note that there are many kinds of generalized inverse $J^{\#}(q)$, and one choice is the weighted generalized-inverse as below [17]

$$J^{\#} = W_J^{-1}J^T(JW_J^{-1}J^T)^{-1} \tag{3}$$

with $W_J \in \mathbb{R}^{n\times n}$ being a selected weighting matrix.

Note that the rank of the null space matrix $N$ is $r = n - m$ which means $N$ is not of full rank and this may cause degradation of the control performance [19]. To deal with this problem, an auxiliary velocity vector $\vartheta$ is introduced by using a joint space decomposition algorithm as follows,

$$\dot{q}_n = N\xi = Z(q)\vartheta \tag{4}$$

where $\dot{q}_n$ is defined in (2), which denotes the null motion, $Z(q)$ is a $n \times r$ matrix, which is of full column rank and chosen to satisfy $J(q)Z(q) = \mathbf{0}$, while $\vartheta \in \mathbb{R}^r$ is a minimal

null-space velocity vector [19].

To obtain the relation between the null-space velocity $\vartheta$ and joint velocity $\dot{q}$, the generalized inverse is employed as below,

$$\vartheta = Z^{\#}(q)\dot{q} \tag{5}$$

where $Z^{\#} \in \mathbb{R}^{r\times n}$ is a weighted generalized inverse matrix of $Z$, which is defined by

$$Z^{\#} = (Z^T W_Z Z)^{-1} Z^T W_Z \tag{6}$$

with $W_Z \in \mathbb{R}^{n\times n}$ being a weighting matrix.

Until now, the inverse kinematics can be represented by combining (2) and (4) as $\dot{q} = J^{\#}(q)\dot{x} + Z(q)\vartheta$. In addition to the kinematics, the robot dynamics can be formulated by using Lagrangian method,

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + \tau_e = \tau \tag{7}$$

where $M(q) \in \mathbb{R}^{n\times n}$ is the inertia/mass matrix, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ and $G(q) \in \mathbb{R}^n$ denote Coriolis and centrifugal matrix and gravity term, respectively. $\tau_e \in \mathbb{R}^n$ is external joint torque while $\tau \in \mathbb{R}^n$ is the joint driving torque. Note that in a human-robot interaction scenario, the external torque $\tau_e$ applied on the joint may be caused by two kinds of forces, i.e., an interaction force $f_e \in \mathbb{R}^m$ applied on the end-effector and a contact force $f_i \in \mathbb{R}^m$ applied on the links, which are depicted in Fig. 1. Here, $\tau_e$ can be represented as a net torque as below [40]

$$\tau_e = J^T(q) f_e + \sum_{i=1}^{k} J_i^T(q) f_i \tag{8}$$

where $f_i$ denotes the $i$th a contact force at the robot body, and $J_i$ is the manipulator Jacobian matrix associated with $i$th contact force.

### *2.2. Impedance Control*

Assume that the robot end-effector is contacting with the environment, for example, a vertical bar connecting with a mass-damping-spring model as depicted in Fig. 1,

$$M_e(t)\ddot{x}(t) + C_e(t)\dot{x}(t) + G_e(t)x(t) = f_e(t) \tag{9}$$

where $M_e \in \mathbb{R}^{m\times m}$, $C_e \in \mathbb{R}^{m\times m}$ and $G_e \in \mathbb{R}^m$ are the inertia matrix, damping matrix and stiffness matrix, respectively, the subscript $e$ denotes the 'environment' and $f_e(t)$ is the force applied on the end-effector. Note the above model is often used to describe the environment in interaction scenarios, e.g., an elastic ball [8] or a human limb in human-robot interactions [1]. Also, note that the model is unknown in a real implementation and it is only used for analysis.

In order to control the robot to interact with the environment, a Cartesian impedance control model is considered as follows

$$M_d(\ddot{x}_d - \ddot{x}_r) + C_d(q,\dot{q})(\dot{x}_d - \dot{x}_r) + G_d(q)(x_d - x_r) = f_e(t) \tag{10}$$

where $M_d \in \mathbb{R}^{m\times m}$ is a selected inertia matrix, $C_d \in \mathbb{R}^{m\times m}$ is a selected damping matrix and $G_d \in \mathbb{R}^m$ is a selected stiffness matrix, $x_d \in \mathbb{R}^m$ is the desired end-effector trajectory, while $\dot{x}_d \in \mathbb{R}^m$ and $\ddot{x}_d \in \mathbb{R}^m$ are the desired velocity and acceleration, respectively. $x_r \in \mathbb{R}^m$ is a reference trajectory in Cartesian Space, $\dot{x}_r \in \mathbb{R}^m$ and $\ddot{x}_r \in \mathbb{R}^m$ are the first and second order derivatives of $x_r$, respectively.

Generally, if the environmental parameters are known accurately, we can obtain the reference trajectory $x_r$ directly from the impedance model (10). However, it is extremely hard to acquire an accurate environmental model in some practical scenarios, e.g., human limbs. Without suitable parameters, the reference trajectory $x_r$ can not be generated appropriately, so the interaction with the unknown environment may fail. In this regard, a learning mechanism is necessary to find the robot's impedance parameters $M_d(t)$, $C_d(t)$ and $G_d(t)$ for obtaining desired interaction performance.

### *2.3. Impedance Learning Scheme*

**Lemma 1.** *Let us consider a linear time-varying system as below [41]*

$$\begin{aligned}\dot{\zeta}(t) &= \mathcal{A}(t)\zeta(t) + \mathcal{B}(t)u(t)\\ r(t) &= \mathcal{C}(t)\zeta(t)\end{aligned} \tag{11}$$

*where $\zeta(t)$ is a state space vector, $u(t)$ is the input of the system, $\mathcal{A}(t)$, $\mathcal{B}(t)$ and $\mathcal{C}(t)$ are time-varying parametric matrices. The control input is chosen to update iteratively as*

$$u^k(t) = u^{k-1}(t) + \boldsymbol{\lambda}[\dot{r}_d(t) - \dot{r}^k(t)] \tag{12}$$

*where $k$ denotes the iteration step, $r_d(t)$ is the desired output and $\boldsymbol{\lambda} \in \mathbb{R}$ is a parameter satisfying $||I - \lambda\mathcal{B}(t)\mathcal{C}(t)||_\infty < 1$, with $I$ being an identity matrix of a proper dimension.*

*If the initial condition is chosen to $r^k(0) = r_d(0)$ with $\det|C(t)B(t)| \neq 0$, then we can conclude from [41] that the system (11) is convergent with $r^k(t) \to r_d(t)$ uniformly in the period $t \in [0, t_f]$ with $k \to \infty$, where $t_f$ is the iteration period.*

To employ the above betterment scheme to learn the impedance parameters, a cost function $\aleph \in \mathbb{R}$ is designed to evaluate the performance of the impedance control. An example of the cost function can be selected as a combination of position tracking errors and interaction forces to regulate the interaction performance.

To make the environmental dynamics (9) be coherent with the linear system form in (11), the environment model (9) can be rewritten as below

$$\begin{aligned}\begin{bmatrix}\dot{\delta}_1\\ \dot{\delta}_2\\ \dot{\delta}_3\end{bmatrix} &= \begin{bmatrix}0 & I & 0\\ -M_e^{-1}(t)G_e(t) & -M_e^{-1}(t)C_e(t) & 0\\ 0 & 0 & 0\end{bmatrix}\\ &\times \begin{bmatrix}\delta_1\\ \delta_2\\ \delta_3\end{bmatrix} + \begin{bmatrix}0\\ -M_e^{-1}\\ I\end{bmatrix} f_e(t)\end{aligned} \tag{13}$$

where $\delta_1(t) = x(t)$, $\delta_2(t) = \dot{x}(t)$ and $\delta_3(t) = \int_0^t f_e(r)dr$. More compactly, the above

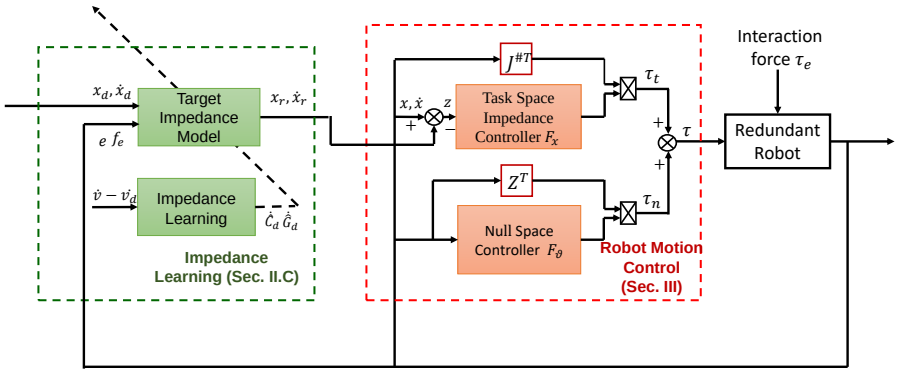


Figure 2: Control block of the overall system

equation can be rewritten as

$$\begin{aligned} \dot{\zeta}(t) &= \mathcal{A}\zeta(t) + \mathcal{B}f_e(t) \\ r(t) &= \mathcal{C}(t)\zeta(t) \end{aligned} \tag{14}$$

where $\zeta = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix}$, $\mathcal{B} = \begin{bmatrix} 0 \\ -M_e^{-1} \\ I \end{bmatrix}$ and $\mathcal{A} = \begin{bmatrix} 0 & I & 0 \\ -M_e^{-1}(t)G_e(t) & -M_e^{-1}(t)C_e(t) & 0 \\ 0 & 0 & 0 \end{bmatrix}$.
$\mathcal{C}$ is a mapping matrix, which is designed according the requirement of the task.

According to (12), the desired interaction force $f_e(t)$ in (14) should be updated by

$$f_e^k(t) = f_e^{k-1}(t) - \lambda(\dot{r}^k(t) - \dot{r}_d(t)) \tag{15}$$

where $\lambda$ is the gain parameter defined in (12), which guarantees $\dot{r}^k(t) \to \dot{r}_d(t)$ when $k \to \infty$. In this way, the error between $r^k(t)$ and $r_d(t)$ can be reduced by iteratively updating $f_e$ in terms of Lemma 1. And it can further converge to zero if $r^k(0)$ equal to $r_d(0)$.

To achieve this objective, here we design a cost function $\boldsymbol{\theta} = ||r(t) - r_d(t)||_2 \in \mathbb{R}$, where $||\cdot||_2$ is 2-norm. Then the iterative law can be designed as

$$f_e^k(t) = f_e^{k-1}(t) - \alpha(\partial\boldsymbol{\theta}^k(t)/\partial f_e^k(t))^T \tag{16}$$

where $\alpha \in \mathbb{R}$ is a selected learning rate. Comparing with (15) and (16), we have

$$\alpha(\partial\boldsymbol{\theta}^k(t)/\partial f_e^k(t))^T = \lambda(\dot{r}^k(t) - \dot{r}_d(t)) \tag{17}$$

This equation gives the relationship between the designed cost function $\boldsymbol{\theta}$ and the output error of $\dot{r}^k$. Also, the partial derivative of (10) with respect to $f_e$ yields

$$\frac{\partial f_e^k(t)^T}{\partial C_d^k(t)} = \dot{e}_x^k, \quad \frac{\partial f_e^k(t)^T}{\partial G_d^k(t)} = e_x^k \tag{18}$$

where $e_x = x_d - x_r$.

Based on the above two equations, here we can design the learning law of the impedance parameters as follows

$$\begin{aligned} C_d^k &= C_d^{k-1}(t) - \alpha_C^{'} \left( \frac{\partial \boldsymbol{\theta}^k(t)}{\partial C_d^k(t)} \right)^T \\ &= C_d^{k-1}(t) - \alpha_C^{'} \left( \frac{\partial f^k(t)}{\partial C_d^k(t)} \right)^T \left( \frac{\partial \boldsymbol{\theta}^k(t)}{\partial f^k(t)} \right)^T \\ &= C_d^{k-1}(t) - \alpha_C \dot{e}_x^k (\dot{r}^k(t) - \dot{r}_d(t)) \\ G_d^k &= G_d^{k-1}(t) - \alpha_G^{'} \left( \frac{\partial \boldsymbol{\theta}^k(t)}{\partial G_d^k(t)} \right)^T \\ &= G_d^{k-1}(t) - \alpha_G^{'} \left( \frac{\partial f^k(t)}{\partial G_d^k(t)} \right)^T \left( \frac{\partial \boldsymbol{\theta}^k(t)}{\partial f^k(t)} \right)^T \\ &= G_d^{k-1}(t) - \alpha_G e_x^k (\dot{r}^k(t) - \dot{r}_d(t)) \end{aligned} \tag{19}$$

where $\alpha_C = \lambda \alpha_C^{'}/\alpha \in \mathbb{R}$, $\alpha_G = \lambda \alpha_G^{'}/\alpha \in \mathbb{R}$, with $\alpha_C^{'}$ and $\alpha_G^{'}$ being the designed learning rates.

**Remark 1.** *The adaptation of impedance parameters in (19) is designed to iteratively adjust the interaction force towards a direction that the cost function $\boldsymbol{\theta}$ can be gradually reduced. Note that in this paper, we only update the parameters $C_d$ and $G_d$, while $M_d$ is chosen as the apparent inertia since an arbitrary $M_d$ may lead to instability of the system.*

**Remark 2.** *The designed cost function $\boldsymbol{\theta} = ||r(t) - r_d(t)||_2$ plays a role to keep the balance between $f_e$ and $e_x$ by updating the impedance parameters. For instance, if we select the cost function as $\boldsymbol{\theta} = ||ae_x - b\int_0^t e_f(s)ds||_2$, $e_f = f_e - f_d$ with $f_d$ being the desired interaction force, while $a$ and $b$ are positive constants, the adaptation is to find a balance between $e_x$ and $\int_0^t e_f(s)ds$ and to make $\boldsymbol{\theta}$ as small as possible. In this respect, the learning algorithm is to provide a stable adaptation of the reference trajectory, to make the interaction force become smaller. However, if the robot end-effector is in contact with an environment surface, there will be always a counterforce applied on the robot, thus the interactive force may not converge to zero.*

## 3. Control Design for the Redundant Robot

### *3.1. Control Objective*

Through the impedance learning in the last section, a group of impedance parameters $M_d$, $C_d$ and $G_d$ can be obtained and the reference trajectory $x_r$, $\dot{x}_r$ and $\ddot{x}_r$ can be calculated from the impedance model (10).

Note that for a redundant robot, the inverse solution of joint velocity is not unique since the DOF of the task space is less than the DOF of joints. In other words, we have multiple solutions of the joint velocity. In this paper, we develop a multi-hierarchy

robot motion controller by employing the null space projection. The control framework of the proposed controller is illustrated in Fig.2. There are mainly two components of the controller, i.e., an outer loop impedance learning controller which responses for learning appropriate impedance parameters, and an inner loop robot motion controller which is able to control the robot with multi-priority tasks. Specifically, the objectives of the robot motion control are twofold:

i) Design a first-priority controller to track a predefined trajectory $x_r$ in task space with learned impedance parameters, such that desired impedance behaviours of the end-effector could be achieved.

ii) Design a second-priority controller in the null space of the main task to provide compliant joint motion under interactive contact force.

### *3.2. Extended Velocity and Decoupled Dynamics*

From equations (2) and (4), the joints' motion is composed of two components, corresponding to the task space velocity and the null space velocity. For ease of the system analysis, an extended velocity vector $\dot{x}_e$ is introduced by including both the task velocity and null space velocity as

$$\dot{x}_e = \begin{bmatrix} \dot{x} \\ \vartheta \end{bmatrix} = J_e \dot{q} = \begin{bmatrix} J(q) \\ Z^{\#}(q) \end{bmatrix} \dot{q} \tag{20}$$

where $J_e(q) = \begin{bmatrix} J(q) \\ Z^{\#}(q) \end{bmatrix}$. Let $J_e^{\#}(q)$ be the generalized inverse of $J_e(q)$ with $J_e^{\#T}(q) = \begin{bmatrix} J^{\#T}(q) \\ Z^T(q) \end{bmatrix}$. It is trivial to have $J_e^{\#}(q)J_e = I$ and $J_e J_e^{\#}(q) J_e = J_e$.

From the equation (20), the mapping between $\dot{x}_e$ and $\dot{q}$ is obtained. The time derivative of (20) yields $\ddot{x}_e = J_e \ddot{q} + \dot{J}_e \dot{q}$. Then, we can decouple the dynamics of the robot (7) into two subsystems, i.e., the task space dynamics and the null space dynamics.

Premultiplying $J_e^{\#T}$ on both sides of (7), and taking $\dot{q} = J^{\#}\dot{x} + Z(q)\vartheta$ into consideration, we can reformulate the system as below

$$\begin{aligned} J_e^{\#T}\tau = \Xi_e(q)\ddot{x}_e + \eta_e(q,\dot{q})\dot{x}_e + J_e^{\#T}G(q) \\ + J_e^{\#T}\tau_e \end{aligned} \tag{21}$$

where

$$\begin{aligned} \Xi_e &= J_e^{\#T} M J_e^{\#} = \begin{bmatrix} J^{\#T}MJ^{\#} & J^{\#T}MZ \\ ZMJ^{\#} & Z^TMZ \end{bmatrix} \\ \eta_e &= \begin{bmatrix} \eta_{11} & \eta_{12} \\ \eta_{21} & \eta_{22} \end{bmatrix} \end{aligned} \tag{22}$$

with

$$\eta_{11}(q,\dot{q}) = (J^{\#T}C(q,\dot{q}) - J^{\#T}MJ^{\#}\dot{J})J^{\#}$$
$$\eta_{12}(q,\dot{q}) = (J^{\#T}C(q,\dot{q}) - J^{\#T}MJ^{\#}\dot{J})Z$$
$$\eta_{21}(q,\dot{q}) = (Z^{T}C(q,\dot{q}) - Z^{T}MZ\dot{Z}^{\#})Z$$
$$\eta_{22}(q,\dot{q}) = (Z^{T}C(q,\dot{q}) - Z^{T}MZ\dot{Z}^{\#})J^{\#}$$

Note that in (22), the matrix $\Xi_e$ is not diagonal, thus the off-diagonal terms are coupled. To control the motion of the task space and null space independently, we need to properly address these terms. Inspired by the work in [18], an inertia-weighted generalized inverse is introduced by selecting $W_J = W_Z = M$ in (3) and (5), such that

$$\begin{cases} J^{\#} = M^{-1}J^{T}(JM^{-1}J^{T})^{-1} \\ Z^{\#} = (Z^{T}MZ)^{-1}Z^{T}M \end{cases} \tag{23}$$

Note $J(q)Z(q) = 0$ according to (4), then we have $J^{\#T}MZ = ZMJ^{\#} = 0$. Thus, $\Xi_e$ becomes

$$\Xi_e = \begin{bmatrix} J^{\#T}MJ^{\#} & 0 \\ 0 & Z^{T}MZ \end{bmatrix} \tag{24}$$

From (24) we can see that $\Xi_e$ is a block diagonal matrix. Thus the dynamics of task space motion and null motion are decoupled in the acceleration level. Then controllers for the two subsystems can be designed individually.

Let us define $\Xi_x = J_e^{\#T}MJ_e^{\#} = (JM^{-1}J^{T})^{-1}$ and $\Xi_\vartheta = Z^{T}MZ$, then the force control terms can be rewritten as bellow,

$$\begin{cases} F_x = \Xi_x\ddot{x}_c + \eta_x + g_x(q) + J^{\#T}\tau_e \\ F_\vartheta = \Xi_\vartheta\dot{\vartheta}_c + \eta_\vartheta + g_\vartheta(q) \end{cases} \tag{25}$$

where $F_x$ denotes the task space control force, $F_\vartheta$ denotes the null space control force with $\begin{bmatrix} F_x \\ F_\vartheta \end{bmatrix} = J_e^{\#T}\tau = \begin{bmatrix} J^{\#T} \\ Z^{T} \end{bmatrix}\tau$, $\ddot{x}_c$, $\dot{\vartheta}_c$ are commanded acceleration trajectories which will be designed later, $\eta_x = \eta_{11}\dot{x} + \eta_{12}Z^{\#}\dot{q}$, $\eta_\vartheta = \eta_{21}\dot{x} + \eta_{22}Z^{\#}\dot{q}$, $g_x(q) = J^{\#T}G(q)$, $g_\vartheta(q) = Z^{T}G(q)$.

### *3.3. Controller Design for the Task Space Motion*

As discussed above, the control goal of the task space motion is to follow a predefined trajectory. Prior to the controller design, error signals are designed as below

$$e = x - x_r \tag{26}$$

$$z = \dot{x} - \dot{x}_s \tag{27}$$

where $\dot{x}_s = \dot{x}_r + K_a e$ with $K_a$ being a positive definite matrix.

The acceleration command $x_c$ is defined as below,

$$\ddot{x}_c = \ddot{x}_s - \Xi_x^{-1}(K_p + \eta_x)z \tag{28}$$

where $K_p$ is a positive definite matrix. Here, $\eta_x$ is introduced to deal with the Coriolis and centrifugal force of the task space.

Substituting (28) into (25), we have the following task space control law

$$F_x = \Xi_x(\ddot{x}_s - \Xi_x^{-1}(K_p + \eta_x)z) + \eta_x + g_x(q) + J^{\#T}\tau_e \tag{29}$$

However, $\tau_e$ may not be available due to the lack of the sensors, which will resulted in the degradation of the control performance. To compensate for the external disturbances and improve the tracking performance, an adaptive disturbance observer $\hat{\tau}_e$ is developed to replace $\tau_e$ in (29). Moreover, the adaptation law is designed as below [42],

$$\dot{\hat{\tau}}_e = \Gamma^{-1} J^{\#} z \tag{30}$$

where $\Gamma$ is a designed positive definite matrix.

**Remark 3.** *Note that the task space tracking errors caused by the interaction forces can also be eliminated by choosing a high gain controller, but this may often lead to oscillation phenomenon. Through estimation of the disturbance, the adaptive disturbance observer can be a more effective method to reduce the tracking errors.*

Combining (21) and (29), we can obtain the error dynamics in the task space as below

$$\Xi_x \dot{z} = -(K_p + \eta_x)z + J^{\#T}\tilde{\tau}_e \tag{31}$$

where $\tilde{\tau}_e = \tau_e - \hat{\tau}_e$.

### *3.4. Controller Design for the Null Space Motion*

In order to accomplish the second control objective, namely, to guarantee a compliant joint motion without affecting the main task when interactive contact forces are acted on the robot, the null space acceleration command is chosen as below

$$\dot{\vartheta}_c = \dot{\vartheta}_d + \Xi_\vartheta^{-1}((K_v + \eta_\vartheta)\tilde{\vartheta} + Z^T K_\phi \tilde{q}) \tag{32}$$

where $\vartheta_d \in \mathbb{R}^r$ is the desired null motion, $\tilde{\vartheta} = \vartheta - \vartheta_d$, $K_v \in \mathbb{R}^{r\times r}$ and $K_\phi \in \mathbb{R}^{n\times n}$ are selected gain matrices, and $\eta_\vartheta$ is designed to deal with the Coriolis and centrifugal torque in the null space.

Substituting (32) into (25), we have

$$\begin{aligned} F_\vartheta &= \Xi_\vartheta(\dot{\vartheta}_d + \Xi_\vartheta^{-1}((K_v + \eta_\vartheta)\tilde{\vartheta} + Z^T K_\phi \tilde{q})) + \eta_\vartheta + g_\vartheta(q) \\ &= \Xi_\vartheta \dot{\vartheta}_d + (K_v + \eta_\vartheta)\tilde{\vartheta} + Z^T K_\phi \tilde{q} + \eta_\vartheta + g_\vartheta(q) \end{aligned} \tag{33}$$

Then the closed-loop dynamics of the robot can be derived based on (21) and (33) as

$$\Xi_\vartheta \dot{\tilde{\vartheta}} + (K_v + \eta_\vartheta)\tilde{\vartheta} + Z^T K_\phi \tilde{q} = Z^T \tau_e \tag{34}$$

We can see from (34) that, although $\Xi_\vartheta$ and $\eta_\vartheta$ are reserved by the robot dynamics, the impedance behaviour of the null space motion can be preserved by selecting proper $K_v$ and $K_\phi$. Note that the external force $\tau_e$ in (8) is composed by two parts, i.e.,

$\tau_{e1} = J^T(q)f_e$ and $\tau_{e2} = \sum_{i=1}^{k} J_i^T(q)f_i$. By multiplying $Z^T(q)$ on both sides of $\tau_{e1}$ and considering $JZ = 0$, we can obtain $Z^T(q)\tau_{e1} = Z^T(q)J^T(q)f_e = 0$. This implies that the force applied in the main task space will not transfer to the null space.

**Remark 4.** *To achieve an appropriate null space motion, the desired null space velocity should be specified. A possible choice is to set the desired null space velocity as zero to allow the robot to maintain the initial configuration. More commonly null space motions can be obtained using an optimization method with specified performance criterion, such as maximized manipulatability or minimized joint effort [43].*

Combining the control commands in (31) and (33) with the definitions of $F_x$ and $F_\vartheta$ in (25), we can rewrite the control torque as

$$\tau = \tau_t + \tau_n \tag{35}$$

with

$$\begin{aligned} \tau_t &= J^T\Big(\Xi_x(\ddot{x}_s - \Xi_x^{-1}(K_p + \eta_x)z) + \eta_x + g_x(q) + J^{\#T}\hat{\tau}_e\Big) \\ \tau_n &= Z^{\#T}\Big(\Xi_\vartheta\dot{\vartheta}_d + (K_v + \eta_\vartheta)\tilde{\vartheta} + Z^T K_\phi\tilde{q} + \eta_\vartheta + g_\vartheta(q)\Big) \end{aligned} \tag{36}$$

From the above analysis, we can see that the control torque consists of two parts, which are responsible for controlling the task space motion and the null space motion. The control block of the overall system is depicted as shown in Fig. 2.

### *3.5. Stability Analysis*

To prove the stability of the hierarchical control framework, a conditional stability analysis was performed in previous studies [43, 44]. In this section, the stability proof is performed based on the conditional stability and the following theorem is presented.

**Theorem 1.** *Given a redundant robot with the dynamics described in (7) and under an assumption of constant (or slowly time-varying) unknown external torque, the control laws proposed in (29), (33) and (35), and the adaptive law in (30) can guarantee the tracking control of the desired task space trajectory $x_r$ under the interaction forces, while providing a complaint behaviour of null motion in joint space when interactive contact forces are acted on the robot body.*

*proof:* Let us consider a Lyapunov function $V_x \in \mathbb{R}$ as

$$V_x = \frac{1}{2}z^T\Xi_x z + \frac{1}{2}\tilde{\tau}_e^T\Gamma\tilde{\tau}_e \tag{37}$$

The time derivative of (37) can be derived as

$$\dot{V}_x = z^T\Xi_x\dot{z} + \frac{1}{2}z^T\dot{\Xi}_x z + \tilde{\tau}_e^T\Gamma\dot{\tilde{\tau}}_e \tag{38}$$

Substituting $(31)$ into $(38)$, we have

$$\dot{V}_x = z^T(-(K_p + \eta_x)z + J^{\#T}\tilde{\tau}_e) + \frac{1}{2}z^T\dot{\Xi}_x z + \tilde{\tau}_e^T\Gamma\dot{\tilde{\tau}}_e \tag{39}$$

during the interaction, by assuming that the interactive contact force is fixed or changed slowly. Substituting the adaptive law (30) into (39) and considering that $\Xi_x - 2\eta_x$ is a skew-symmetric matrix, then we can derive

$$\dot{V}_x = -z^T K_p z - \tilde{\tau}_e^T J^{\#} z + z^T J^{\#T} \tilde{\tau}_e \tag{40}$$

Note that $\tilde{\tau}_e^T J^{\#} z = ((\tilde{\tau}_e^T J^{\#} z)^T)^T = (z^T (\tilde{\tau}_e^T J^{\#})^T)^T = z^T J^{\#T} \tilde{\tau}_e$, where the property $a = a^T$ holds for a given constant scalar $a$. Therefore, the terms $\tilde{\tau}_e^T J^{\#} z$ and $z^T J^{\#T} \tilde{\tau}_e$ are cancelled by each other in (40) and we have that

$$\dot{V}_x = -z^T K_p z \leq 0 \tag{41}$$

From (41) we can see that $\dot{V}_x$ is negative unless $z$ equals to zero. This means that $V_x$ will decrease until $z = 0$. According to LaSalle's invariance principle [45], we can derive that $z$ is asymptomatically stable, and $\tilde{\tau}_e$ is uniformly ultimately bounded. According to the definition of $z$, we can also obtain that $e = 0$ and $\dot{e} = 0$, which imply that tracking of the end-effector position is achieved.

A conditional stability is employed to prove the stability of the null space motion control. In a subset of $\mathcal{C} = \{\tilde{q}, \tilde{\vartheta}, \tilde{\tau}_e, z = 0, e = 0\}$, a positive definite Lyapunov candidate $V_{\mathcal{C}}$ is considered as follows,

$$V_\vartheta = \frac{1}{2}\tilde{\vartheta}^T \Xi_\vartheta(q)\tilde{\vartheta} + \frac{1}{2}\tilde{q}^T K_\phi \tilde{q} + \frac{1}{2}\tilde{\tau}_e^T \Gamma \tilde{\tau}_e \tag{42}$$

Taking the time derivative of (42), we have

$$\dot{V}_\vartheta = \tilde{\vartheta}^T \Xi_\vartheta(q)\dot{\tilde{\vartheta}} + \frac{1}{2}\vartheta^T \dot{\Xi}_\vartheta(q)\vartheta + \tilde{q}^T K_\phi \dot{\tilde{q}} + \tilde{\tau}_e^T \Gamma \dot{\tilde{\tau}}_e \tag{43}$$

Substituting (34) into (43), we have

$$\begin{aligned} \dot{V}_\vartheta = \tilde{\vartheta}^T \Big( & (-(K_v + \eta_\vartheta)\tilde{\vartheta} - Z^T K_\phi \tilde{q} + Z^T \tau_e \Big) \\ & + \frac{1}{2}\vartheta^T \dot{\Xi}_\vartheta(q)\vartheta + \tilde{q}^T K_\phi \dot{\tilde{q}} + \tilde{\tau}_e^T \Gamma \dot{\tilde{\tau}}_e \end{aligned} \tag{44}$$

Having reminded that $\dot{\Xi}_\vartheta - 2\eta_\vartheta$ is skew-symmetric, while $\dot{\tilde{\tau}}_e = 0$ in terms of $\dot{\tilde{\tau}}_e = \Gamma^{-1} J^{\#} z$ and $z = 0$, we can obtain

$$\begin{aligned} \dot{V}_\vartheta = & -\tilde{\vartheta}^T K_\vartheta \tilde{\vartheta} - \tilde{\vartheta}^T Z^T K_\phi \tilde{q} \\ & + \tilde{q}^T K_\phi \dot{\tilde{q}} + \tilde{\vartheta}^T Z^T \tau_e \end{aligned} \tag{45}$$

From $\dot{\tilde{q}} = J^{\#T} e + Z\tilde{\vartheta}$ and $e = 0$ in the subset $\mathcal{C}$, we have that $\dot{\tilde{q}} = Z\tilde{\vartheta}$. Then, (45) can be derived as

$$\dot{V}_\vartheta = -\tilde{\vartheta}^T K_\vartheta \tilde{\vartheta} + \tilde{\vartheta}^T Z^T \tau_e \tag{46}$$

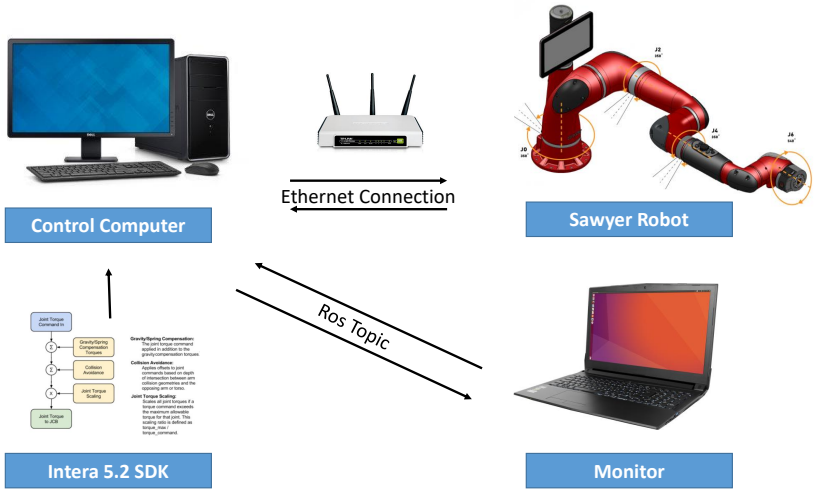


Figure 3: An overview of the experimental set-up

Here, the stability of the system can be analysed in two cases, $Z^T\tau_e = 0$ and $Z^T\tau_e \neq 0$. In the case $Z^T\tau_e = 0$, (46) can be written to

$$\dot{V}_\vartheta = -\tilde{\vartheta}^T K_\vartheta \tilde{\vartheta} \leq 0. \tag{47}$$

Thus, $\vartheta$ is asymptotically stable in the subset $\mathcal{C}$ in terms of the Lasalle's theorem. This implies the convergence of all system states into an invariant set in $\{z = 0, e = 0, \tilde{\vartheta} = 0, Z^T K_\phi \tilde{q} s 0\}$. According to the above mentioned discussion, we can conclude that system variables $e$, $z$ and $\vartheta$ are asymptotically stable, which shows that both the first priority task space control goal and the second priority null space control goal are achieved. This completes the proof.

**Remark 5.** *With the above stability analysis, it is possible to prove asymptotic convergence to zero of the task-space error and the force estimation error. However, this requires the external torque applied on robot body to be a slowly time-varying vector, which may be not easy to achieve in a real experimental environment condition. Nevertheless, the stable tracking performance can still be guaranteed in a small neighbourhood around zero of the tracking errors as long as the external force $\tau_e$ is bounded [46].*

## 4. Experimental Studies

### *4.1. Experimental Set-up*

To further investigate the effectiveness of the proposed impedance learning and control algorithm, experimental studies are conducted on a Sawyer robot as shown in

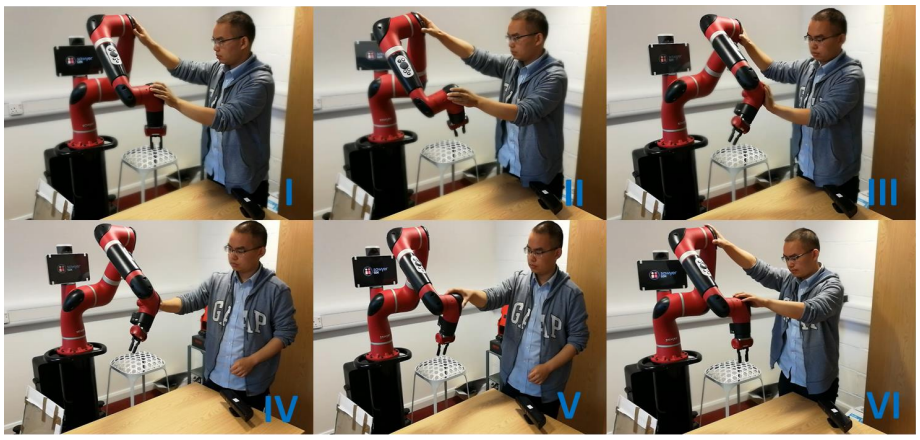


Figure 4: A screenshot of the experiment

Fig. 3. The Sawyer robot is a collaborative robot developed with built-in software development kits (SDK) and sensitive torque/position sensors, which allow the robot to work safely around people.

An overview of the experimental set-up is depicted in Fig. 3. The robot platform is a redundant robot which equips with 7 series elastic actuators (SEA) joints providing torques maximumly at $[80, 80, 40, 40, 9, 9, 9]$ Nm with respect to each joint. 3 joints are equipped on the wrist, 2 joints are on the elbow and 2 joints are on the shoulder. The controller runs in a server computer (Z230, HP, US) with a 3.5Ghz$\times$8 XEON CPU, a 16 GB memory and a 500 GB hard disk. We develop the controller by Robot Operating System (ROS) under the Ubuntu system, with a Sawyer intera SDK provided by the manufacturer. In the experiment, the controller sends control torque to the robot through Ethernet connection at a frequency of 500Hz.

### *4.2. Verification of the Robot Motion Controller*

In order to verify the tracking control performance of our proposed controller in the presence of interactive contact forces, two experimental cases are conducted on the Sawyer robot, i.e., a set-point regulation case and a trajectory tracking case.

#### *4.2.1. Set-point regulation case*

In the first experimental case, the robot end-effector is controlled to maintain at a given position in the Cartesian space, while an interactive contact force is applied by human to push or to pull the elbow of the robot during the control period. The proposed null space controller with consistent generalized inverse is employed to ensure the compliant interaction in joint space without disturbing the end-effector motion, while the adaptive disturbance observer is employed to improve the control accuracy. The end-effector is controlled in $x, y, z$ directions such that 4 DOFs remain for the redundancy of the joints. In this experiment, the control parameters are chosen to be $K_p = [1000, 1000, 1000, 50, 50, 50]$ and $B_p = [30, 30, 30, 2.8, 2.8, 2.8]$, $K_v = 10I$,

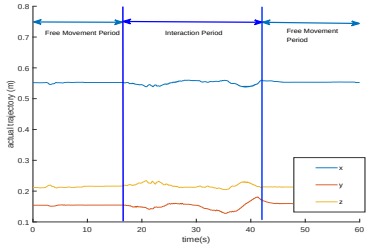


Figure 5: The tracking trajectory of in $x$, $y$ and $z$ direction

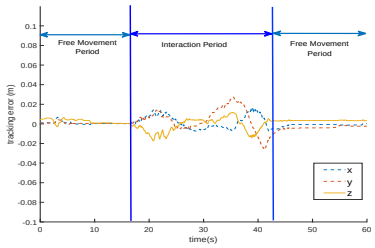


Figure 6: The tracking errors in $x$, $y$ and $z$ direction

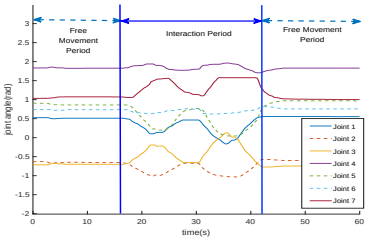


Figure 7: The profiles of joint positions

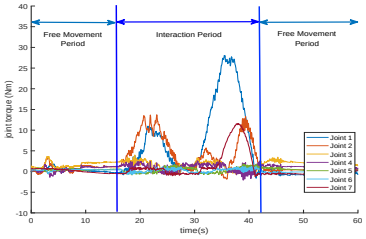


Figure 8: The profiles of control torque $\tau$

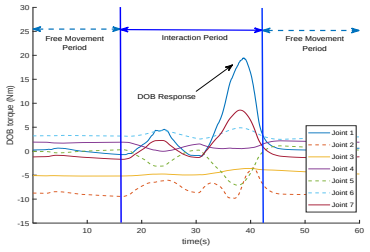


Figure 9: The response of the disturbance observer

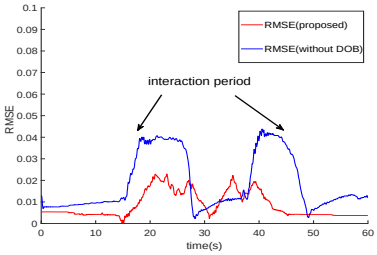


Figure 10: The RMSE of the tracking errors

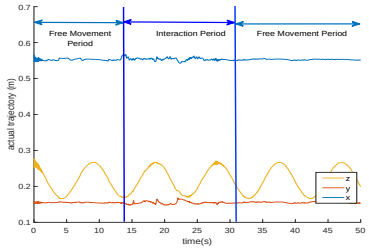


Figure 11: The tracking trajectory performance of the robot

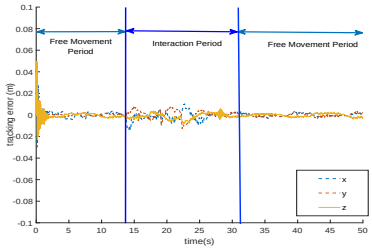


Figure 12: The tracking errors of the robot

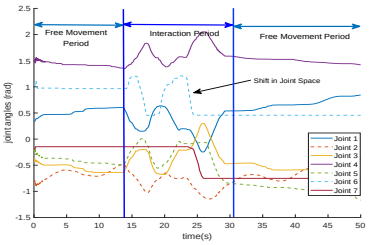


Figure 13: The profile of the joint angles

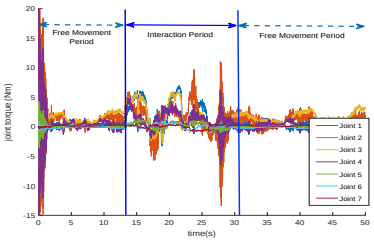


Figure 14: The profile of the control torque

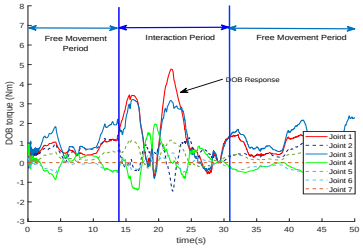


Figure 15: The output of the disturbance observer

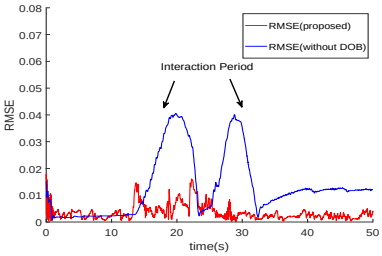


Figure 16: The RMSE of the tracking errors

$B_v = 0.4I$. In this experimental case, the robot end-effector is controlled to be at a desired position of $x_d = [0.552, 0.154, 0.212]$, and $\dot{x}_d = [0, 0, 0]$. The initial joint position is $q_d = [0.52, -0.63, -0.71, 1.83, 0.91, 0.74, 1.02]$ which is kinematically compatible with the end-effector position. Screenshots of the experiment are shown in Fig.4, from which we can see that interactive contact forces are applied at a point close to the elbow of the robot by a human user.

The experimental results are depicted in Figs. 5-9. The interactive contact forces are applied at the time of 17s and last for about 20s to drive the robot away from the initial position. From Fig. 5 we can see that the end-effector is holding at a fixed position in all directions in the free movement period. Also, the tracking errors almost converge to zeros as observed in Fig. 6. In the interaction period, the interactive contact forces are applied. We can see that the robot is able to maintain the end-effector position and only slight movements are observed. The profiles of all robot joints are

depicted in Fig. 7 as well as the control torques in Fig.8, from which we can see that the impacts of the interactive contact force are prominent. Both the joint position and the control torque remain almost steady during the free movement period, but change obviously during the interaction period. Additionally, we can see two peaks in Fig.8, which shows that the robot arm is pushed by external forces during the the interaction period. However, although the joint positions have changed largely, the end-effector can still maintain at a fixed position. The responses of the disturbance observer (DOB) are presented in Fig. 9. To further show the effectiveness of the proposed controller, comparative experimental studies are carried out based on the proposed controller and a controller in [47], where a conventional null space controller is employed without using the DOB. A root mean square error (RMSE) is introduced to evaluated the tracking performance as $\phi = \sqrt{\sum_{i=1}^{3}(x(i) - x_d(i))^2/3}$. The comparative results are depicted in Fig.10, where the RMSE profile of the proposed controller is demonstrated by the red line, and the comparative controller by the blue line. We can see that RMSE of the proposed controller is much smaller than the comparative controller in [47]. For the proposed controller, the maximum RMSE value is 0.0303, which is half smaller than the comparative controller.

*4.2.2. Trajectory tracking case*

In the second experimental case, the robot is controlled to follow a trajectory in the Cartesian space as $x_d = [0.435, 0.135, 0.22{+}0.05\sin(0.3t)]$, $\dot{x}_d = [0, 0, 0.015\cos(0.3t)]$. Note that the robot end-effector is commanded to track a sinusoidal trajectory in $z$ direction while the commands in $x$ and $y$ directions are set to fixed constants. The control parameters are chosen the same as in the first experimental case. The interactive contact forces are applied to the robot during the tracking procedure. In this case, the proposed robot controller should guarantee that the tracking in the main task will not be disturbed by the interactive contact force. The experimental results are depicted in Figs.11-16. From Fig.11 we can see that the end-effector follows the reference trajectory well. The tracking errors are very close to zero during the free movement period, while remaining as a small value during the interaction period as shown in Fig. 12. Figs.13-14 show the profile of the joint angles and control torques. We can see that during the interaction period, the joint angles apparently changed under the interactive contact forces. At the same time, control torques also increase significantly in response to the interactive contact forces. The outputs of the DOB are depicted in Fig.15, where we can see that the disturbance observer has successfully predicted and compensated for the external disturbances. Comparative studies are also carried out based the same controllers as in the set-point regulation case. As shown in Fig. 16, the RMSE of the proposed controller is smaller than the comparative controller.

To further show its effectiveness, we conduct the experiment with a number of trials, to assess the performance of the disturbance observer. The upper subfigures in Figs. 17-19 show the tracking performance in the Cartesian space, while the lower subfigures show the tracking errors with and without using the adaptive disturbance observer. We can see that, the proposed control scheme ensures consistent and satisfactory tracking performance, and tracking errors are much smaller than that without using the adaptive disturbance observer.

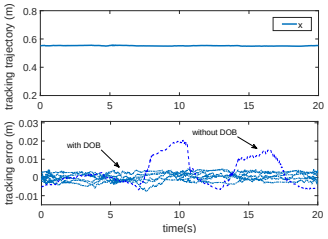


Figure 17: The tracking performance with and without using disturbance observer

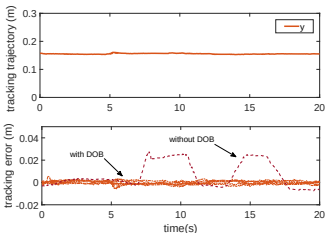


Figure 18: The tracking performance with and without using disturbance observer

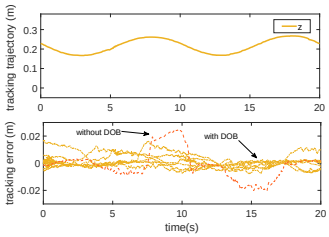


Figure 19: The tracking performance with and without using disturbance observer

### *4.3. Impedance Learning Results*

In this section, we verify the validity of the proposed impedance learning scheme based on both numerical simulations and experiments.

#### *4.3.1. Simulations*

In the simulation, a 3 DOF redundant robot is employed to track a desired trajectory in a vertical plane using the proposed impedance control framework. The robot contacts with the environment such that there are reaction forces exerted on the robot end-effector. The objective of the impedance learning is to achieve the desired performance on interaction forces and tracking errors. To simplify the learning process, we only employ the impedance control in $x$ direction and the motion in $y$ direction is set to a fixed value, such that only one dimensional impedance parameter is considered.

The simulated environmental model is described in (9) with the parameters chosen to be $M_e(t) = 0.0$, $C_e(t) = 1.0$ and $G_e(t) = 10.0$. Note that environmental parameters are unknown for users and they are only used for the simulation purpose. The desired trajectories are chosen to be $x_d = 0.2 - 0.2e^{-t}, \dot{x}_d = 0.2e^{-t}$. To control the interaction motion, impedance control is used to track the desired trajectory while to maintain the interaction forces. The initial impedance control parameters are selected to be $C_d^0 = 15$, $G_d^0 = 10$. The gain parameters $\alpha_C$ and $\alpha_G$ are selected to $\alpha_C = 2$ and $\alpha_G = 2$.

To achieve a balance between the interaction force and tracking errors, the cost function is designed to be $\boldsymbol{\theta}^k(t) = ||10e_x(t) - 3\int_0^t e_f(s)ds||_2$ with $\mathcal{C} = [10, 0, 3]$ and $f_d = 0$. Then, iterative learning is employed to find a minimum cost function

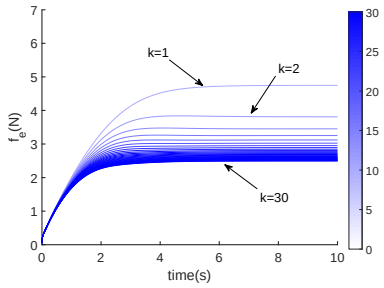


Figure 20: The evolution of the interaction force $f_e$

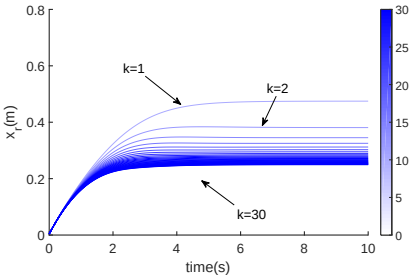


Figure 21: The evolution of the reference trajectory $x_r$

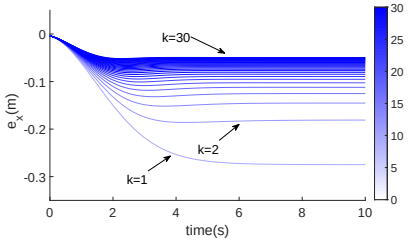


Figure 22: The evolution of the tracking error $e$

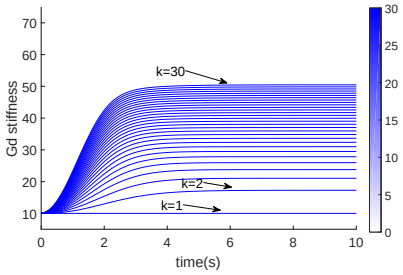


Figure 23: The evolution of the stiffness parameter $G_d$

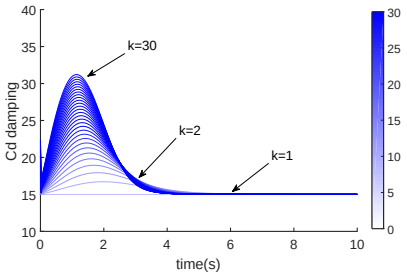


Figure 24: The evolution of the damping parameter $C_d$

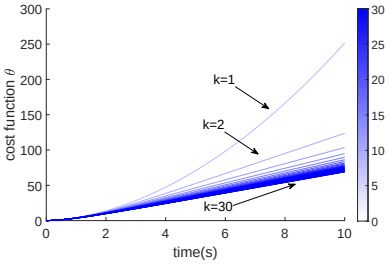


Figure 25: The evolution of the cost function $\boldsymbol{\theta}$

to optimize the tracking performance. Simulation results are performed as shown in Figs.20-25 with 30 iterations. For each figure, there is a colour bar on the right hand side, which shows that the colour of the result curves shifts from light blue to dark blue as the iteration number increases.

As seen from Fig.20, the interaction force decreases when the iteration number increases and finally converges to a value near $2N$. Additionally, the tracking errors become smaller when the iteration number increases, and converge to a small neighbourhood near zero, as shown in Fig.22. The iterative learning profiles of the stiffness parameter $G_d$ and damping parameter $C_d$ are depicted in Fig.23 and Fig.24, respectively. As shown in Fig.23, the stiffness is initialized as $G_d(k = 1) = 10$ and then increases gradually as the iteration number increases. For most of the iterations, the stiffness increases in the first 3 seconds then converges in the rest of the time. Also, the damping parameter achieves its peak value in the period of the 1-2 seconds in most of the iterations, and finally converges to the initial value in 5-10 seconds. This is reasonable since the desired trajectory $x_d$ is a saturation function, which rises in the first 3 seconds then converges to a fixed value in the rest of the time. Therefore, the stiffness $G_d$ becomes larger to result in a smaller interaction force in the beginning. On the other hand, the increase of damping could result in a smaller value of $\dot{x}_r$ as the damping parameter is related to the velocity. When the desired trajectory $x_d$ saturates and the robot stops moving, the damping parameter has reduced to a small value to obtain a smaller interaction force. From Fig.25 we can further observe that the cost function decreases significantly when the iteration number increases, reaching about 5 times less than its first iteration. Hence, the above results have illustrated that the proposed impedance learning framework is effective to iteratively update the impedance parameters to adapt to the environment interaction force.

*4.3.2. Experiments*

The impedance learning experiment is performed on the Sawyer robot with the proposed robot motion controller which has been verified in Section 4.2. The experiment set-up is depicted in Fig. 26. In this experiment, the Sawyer robot is controlled to touch a flexible cardboard fixed on the table as shown in Fig. 26. The robot end-effector is initialized at about 30cm above the blue board, then follows a straight line to touch the blue board. When the robot contacts the blue board, the end-effector will continue to move for a short distance, which results in counteracting forces on the end-effector. The forces can be collected by the built-in torque sensors of Sawyer robot. In this experiment, totally 20 trials are conducted to learn the impedance parameters and the multi-priority control scheme is adopted to control the robot.

The desired trajectory in the vertical direction ($z$ direction) is given as $x_d = 0.031 + 0.3e^{-t}$ with $t = 10$, which is also depicted in Fig.27. In this way, the robot can quickly move close to the board in the first 5 seconds, then contacts with it in the next 5 seconds. The cost function is chosen as $\boldsymbol{\theta}^k(t) = ||20e_x(t) - 3\int_0^t e_f(s)ds||_2$. The proposed impedance learning algorithm is employed with 20 iterations.

The experimental results are depicted in Figs. 28-32. Fig. 28 shows the evolution of the integration of the interaction forces, which can be observed increasing rapidly after contact with the surface, and reaching about 37N at the end of the first iteration. When the iteration number increased, the profile of the interaction force becomes lower than

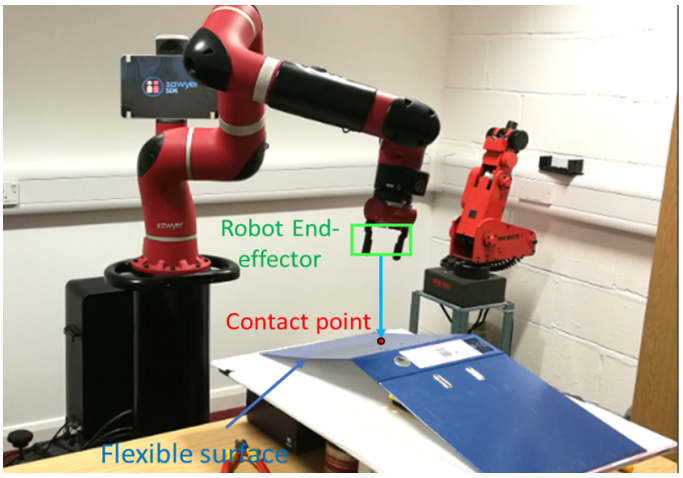


Figure 26: The experiment set up of the impedance learning

the previous iteration and finally reduces to about 27N at the last interaction. Figs.29-30 show the evolution of the stiffness and damping parameters and Fig.31 shows the evolution of the reference trajectory. Both the stiffness parameters and damping parameters are iteratively increased when the iteration number increases. The evolution of the reference trajectory is depicted in Fig. 31, from which we can see that the reference trajectory of the 20th iteration is about 1.5cm higher than the 1st iteration, which implies the reference trajectory is successfully updated to reduce the reaction forces. Besides, the evolution of the cost function depicted in Fig.32 shows that the cost function decreases iteratively. Fig. 28 and Fig. 32 have verified the effectiveness of the proposed learning algorithm, where the decrease of the cost function and interaction force are presented. As we can also see in Fig. 29 and Fig. 30, in the last a few iterations, the added values of both stiffness and damping parameters obviously decrease. Thus, convergence of these parameters can be achieved.

## 5. Conclusion

This paper develops an interaction control framework for redundant manipulators to provide compliance in the robot joint space, such that compliant physical interaction is ensured without affecting the main task. To achieve this goal, we develop two sub-controllers, i.e., a task space impedance controller, and a null space motion controller. The impedance learning algorithm is employed to adjust the impedance parameters of the task space controller without using a priori environment information, and desired interaction performance is achieved with a lower level of interactive contact force. Moreover, an adaptive disturbance observer is employed to compensate for the external disturbance to improve the tracking accuracy. The experimental results on a 7 DOF Sawyer robot show that the learning framework is able to adjust the target impedance model according to the environmental force while the task execution performance is enhanced when other interaction forces are applied on the robot body.

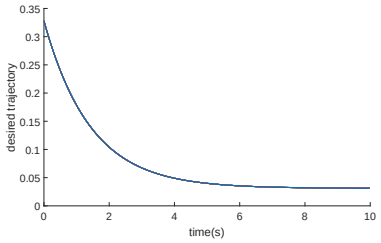


Figure 27: The evolution of the desired trajectory

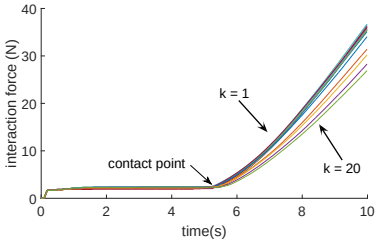


Figure 28: The evolution of the interaction force

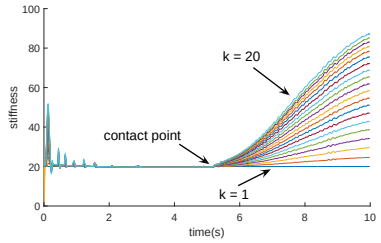


Figure 29: The evolution of the stiffness parameters

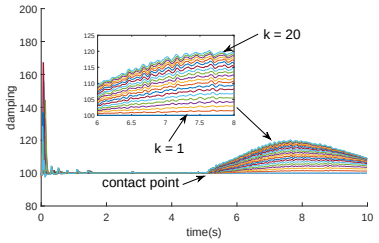


Figure 30: The evolution of the damping parameters

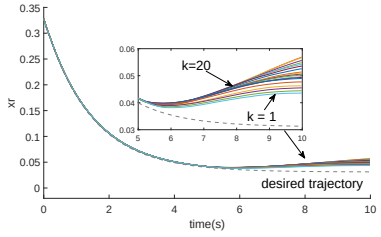


Figure 31: The evolution of the reference trajectory

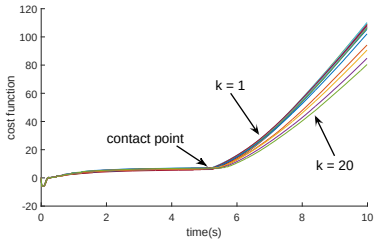


Figure 32: The evolution of the cost function $\boldsymbol{\theta}$

**Acknowledgement**

This work was supported in part by National Nature Science Foundation (NSFC) under Grants 61733004, 61971071 and 51575412, Science and Technology Project of Hunan Province under Grants 2017XK2102, 2018GK2022 and 2018JJ3079, National Key Research and Development Program of China Under Grant 2018YFB1308200, Emergency Project Against the Epidemic of Novel Coronavirus Pneumonia from Special Fund of Hunan Province under Grant 2020SK3007, Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/S001913, and Independent Research Subject Funding from Hunan University State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body.

[1] C. Yang, G. Ganesh, S. Haddadin, S. Parusel, A. Albu-Schaeffer, E. Burdet, Human-like adaptation of force and impedance in stable and unstable interactions, IEEE transactions on robotics 27 (5) (2011) 918–930.

[2] Y. Li, S. S. Ge, C. Yang, X. Li, K. P. Tee, Model-free impedance control for safe human-robot interaction, in: Robotics and Automation (ICRA), 2011 IEEE International Conference on, IEEE, 2011, pp. 6021–6026.

[3] K. Kong, J. Bae, M. Tomizuka, Control of rotary series elastic actuator for ideal force-mode actuation in human–robot interaction applications, IEEE/ASME transactions on mechatronics 14 (1) (2009) 105–118.

[4] Y. Li, G. Carboni, F. Gonzalez, D. Campolo, E. Burdet, Differential game theory for versatile physical human–robot interaction, Nature Machine Intelligence 1 (1) (2019) 36–43.

[5] J. Huang, W. Huo, W. Xu, S. Mohammed, Y. Amirat, Control of upper-limb power-assist exoskeleton using a human-robot interface based on motion intention recognition, IEEE transactions on automation science and engineering 12 (4) (2015) 1257–1270.

[6] J. Luo, Z. Lin, Y. Li, C. Yang, A teleoperation framework for mobile robots based on shared control, IEEE Robotics and Automation Letters 5 (2) (2019) 377–384.

[7] Y. Li, X. Zhou, J. Zhong, X. Li, Robotic impedance learning for robot-assisted physical training, Frontiers in Robotics and AI 6 (2019) 78.

[8] C. Yang, K. Huang, H. Cheng, Y. Li, C.-Y. Su, Haptic identification by elm-controlled uncertain manipulator, IEEE Transactions on Systems, Man, and Cybernetics: Systems.

[9] Y. Li, G. Ganesh, N. Jarrassé, S. Haddadin, A. Albu-Schaeffer, E. Burdet, Force, impedance, and trajectory learning for contact tooling and haptic identification, IEEE Transactions on Robotics 34 (5) (2018) 1170–1182.

[10] C. Yang, Y. Jiang, Z. Li, W. He, C.-Y. Su, Neural control of bimanual robots with guaranteed global stability and motion precision, IEEE Transactions on Industrial Informatics 13 (03) (2017) 1162–1171.

[11] Y. Li, S. S. Ge, Impedance learning for robots interacting with unknown environments, IEEE Transactions on Control Systems Technology 22 (4) (2014) 1422–1432.

[12] H. Su, W. Qi, C. Yang, A. Aliverti, G. Ferrigno, E. De Momi, Deep neural network approach in human-like redundancy optimization for anthropomorphic manipulators, IEEE Access 7 (2019) 124207–124216.

[13] A. Dietrich, C. Ott, A. Albu-Schäffer, An overview of null space projections for redundant, torque-controlled robots, The International Journal of Robotics Research 34 (11) (2015) 1385–1400.

[14] J. Wu, J. Wang, L. Wang, T. Li, Dynamics and control of a planar 3-dof parallel manipulator with actuation redundancy, Mechanism and Machine Theory 44 (4) (2009) 835–849.

[15] J. Wu, J. Wang, Z. You, An overview of dynamic parameter identification of robots, Robotics and computer-integrated manufacturing 26 (5) (2010) 414–419.

[16] J. Wu, G. Yu, Y. Gao, L. Wang, Mechatronics modeling and vibration analysis of a 2-dof parallel manipulator in a 5-dof hybrid machine tool, Mechanism and Machine Theory 121 (2018) 430–445.

[17] Y. Oh, W. Chung, Y. Youm, Extended impedance control of redundant manipulators based on weighted decomposition of joint space, Journal of Field Robotics 15 (5) (1998) 231–258.

[18] B. Nemec, L. Zlajpah, Force control of redundant robots in unstructured environment, IEEE transactions on industrial electronics 49 (1) (2002) 233–240.

[19] B. Nemec, L. Žlajpah, D. Omrčen, Comparison of null-space and minimal null-space control algorithms, Robotica 25 (5) (2007) 511–520.

[20] C. Ott, A. Dietrich, A. Albu-Schäffer, Prioritized multi-task compliance control of redundant manipulators, Automatica 53 (2015) 416–423.

[21] F. Ficuciello, R. Carloni, L. C. Visser, S. Stramigioli, Port-hamiltonian modeling for soft-finger manipulation, in: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, IEEE, 2010, pp. 4281–4286.

[22] C. Ott, Cartesian impedance control: The rigid body case, in: Cartesian Impedance Control of Redundant and Flexible-Joint Robots, Springer, 2008, pp. 29–44.

[23] A. Albu-Schäffer, C. Ott, G. Hirzinger, A unified passivity-based control framework for position, torque and impedance control of flexible joint robots, The international journal of robotics research 26 (1) (2007) 23–39.

[24] B. Kim, J. Park, S. Park, S. Kang, Impedance learning for robotic contact tasks using natural actor-critic algorithm., IEEE Trans. Systems, Man, and Cybernetics, Part B 40 (2) (2010) 433–443.

[25] J. Buchli, F. Stulp, E. Theodorou, S. Schaal, Learning variable impedance control, The International Journal of Robotics Research 30 (7) (2011) 820–833.

[26] B. Corteville, E. Aertbeliën, H. Bruyninckx, J. De Schutter, H. Van Brussel, Human-inspired robot assistant for fast point-to-point movements, in: Robotics and Automation, 2007 IEEE International Conference on, IEEE, 2007, pp. 3639–3644.

[27] C. Wang, Y. Li, S. S. Ge, T. H. Lee, Reference adaptation for robots in physical interactions with unknown environments, IEEE transactions on cybernetics 47 (11) (2017) 3504–3515.

[28] J. Fu, T.-F. Li, T. Chai, C.-Y. Su, Sampled-data-based stabilization of switched linear neutral systems, Automatica 72 (2016) 92–99.

[29] L. Ding, S. Li, Y.-J. Liu, H. Gao, C. Chen, Z. Deng, Adaptive neural network-based tracking control for full-state constrained wheeled mobile robotic system, IEEE Transactions on Systems, Man, and Cybernetics: Systems 47 (8) (2017) 2410–2419.

[30] L. Ding, S. Li, H. Gao, Y.-J. Liu, L. Huang, Z. Deng, Adaptive neural network-based finite-time online optimal tracking control of the nonlinear system with dead zone, IEEE Transactions on Cybernetics.

[31] J. Fu, R. Ma, T. Chai, Adaptive finite-time stabilization of a class of uncertain nonlinear systems via logic-based switchings, IEEE Transactions on Automatic Control 62 (11) (2017) 5998–6003.

[32] W.-H. Chen, Disturbance observer based control for nonlinear systems, IEEE/ASME transactions on mechatronics 9 (4) (2004) 706–710.

[33] J. Huang, S. Ri, L. Liu, Y. Wang, J. Kim, G. Pak, Nonlinear disturbance observer-based dynamic surface control of mobile wheeled inverted pendulum, IEEE Transactions on Control Systems Technology 23 (6) (2015) 2400–2407.

[34] J. Huang, S. Ri, T. Fukuda, Y. Wang, A disturbance observer based sliding mode control for a class of underactuated robotic system with mismatched uncertainties, IEEE Transactions on Automatic Control 64 (6) (2018) 2480–2487.

[35] W. He, Z. Yan, C. Sun, Y. Chen, Adaptive neural network control of a flapping wing micro aerial vehicle with disturbance observer, IEEE Transactions on Cybernetics 47 (10) (2017) 3452–3465.

[36] M. Chen, S. S. Ge, Adaptive neural output feedback control of uncertain nonlinear systems with unknown hysteresis using disturbance observer, IEEE Transactions on Industrial Electronics 62 (12) (2015) 7706–7716.

[37] J. Huang, M. Zhang, S. Ri, C. Xiong, Z. Li, Y. Kang, High-order disturbance-observer-based sliding mode control for mobile wheeled inverted pendulum systems, IEEE Transactions on Industrial Electronics 67 (3) (2019) 2030–2041.

[38] J.-S. Han, T.-I. Kim, T.-H. Oh, Y.-S. Kim, J.-H. Lee, S.-O. Kim, S.-S. Lee, S.-H. Lee, et al., Error-dynamics-based performance shaping methodology for discrete-time sliding mode control with disturbance observer, IFAC-PapersOnLine 52 (15) (2019) 460–464.

[39] J.-S. Han, T.-I. Kim, T.-H. Oh, S.-H. Lee, D.-i. D. Cho, Effective disturbance compensation method under control saturation in discrete-time sliding mode control, IEEE Transactions on Industrial Electronics, to be published.

[40] R. Platt, M. Abdallah, C. Wampler, Multiple-priority impedance control, in: Robotics and Automation (ICRA), 2011 IEEE International Conference on, IEEE, 2011, pp. 6033–6038.

[41] R. Ikeura, H. Inooka, Variable impedance control of a robot for cooperation with a human, in: Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on, Vol. 3, IEEE, 1995, pp. 3097–3102.

[42] W.-H. Chen, Nonlinear disturbance observer-enhanced dynamic inversion control of missiles, Journal of Guidance, Control, and Dynamics 26 (1) (2003) 161–166.

[43] H. Sadeghian, L. Villani, M. Keshmiri, B. Siciliano, Task-space control of robot manipulators with null-space compliance, IEEE Transactions on Robotics 30 (2) (2014) 493–506.

[44] C. Ott, A. Kugi, Y. Nakamura, Resolving the problem of non-integrability of nullspace velocities for compliance control of redundant manipulators by using semi-definite lyapunov functions, in: Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, IEEE, 2008, pp. 1999–2004.

[45] J. P. LaSalle, The stability and control of discrete processes, Vol. 62, Springer Science & Business Media, 2012.

[46] J.-J. E. Slotine, W. Li, et al., Applied nonlinear control, Vol. 199, Prentice hall Englewood Cliffs, NJ, 1991.

[47] J. Park, O. Khatib, Robot multiple contact control, Robotica 26 (5) (2008) 667–677.